Claims

1. A method for cryptographically converting a digital input block into a digital output block; said conversion comprising the step of merging a selected part M1 of said digital input block with a first key K1 and producing a data block B1 which non-linearly depends on said selected part M1 and said first key K1, and where a selected part of said digital output block is derived from said data block B1,
characterised in that said merging step is performed by executing a non-linear function g for non-linearly merging said selected part M1 and said first key K1 in one, sequentially inseparable step.

2. A method as claimed in claim 1, wherein said method comprises the steps of:

- splitting said digital input block into said selected part M1 and a second part M2 before executing said merging step;
- executing a non-linear function g^{-1} to merge said second block M2 with a second key K2 in one, sequentially inseparable step, producing a data block B2 as output; said non-linear function g^{-1} being the inverse of said non-linear function g; and
- forming combined data from data in said data block B1 and in said data block B2; said digital output block being derived from said combined data.

3. A method as claimed in claim 1, wherein said merging step comprises the steps of:

- splitting said selected part M1 in a first plurality n of sub-blocks $m_0,..,m_{n-1}$ of substantially equal length;
- splitting said first key K1 in said first plurality n of sub-keys $k_0,..,k_{n-1}$, substantially having equal length, the sub-key k_i corresponding to the sub-block m_i, for $i = 0$ to n-1; and
- separately processing each of said sub-blocks m_i by executing for each of said sub-blocks m_i a same non-linear function h for non-linearly merging a sub-block b_i derived from said sub-block m_i with said corresponding sub-key k_i in one, sequentially inseparable

step and producing said first plurality of output sub-blocks $h(b_i, k_i)$; and

- combining sub-blocks t_i derived from said first plurality of said output sub-blocks $h(b_i, k_i)$ to form said data block B1.

4. A method as claimed in claim 2 and 3, wherein said step of executing said non-linear function g^{-1} comprises the steps of:

- splitting said second part M2 in said first plurality n of sub-blocks $m_n,..,m_{2n-1}$, substantially having equal length;

- splitting said key K2 in said first plurality n of sub-keys $k_n,..,k_{2n-1}$, substantially having equal length, the sub-key k_i corresponding to the sub-block m_i, for $i = n$ to $2n-1$;

- executing for each of said sub-blocks m_i a same non-linear function h^{-1} for non-linearly merging a sub-block b_i derived from said sub-block m_i with said corresponding sub-key k_i and producing said first plurality of an output sub-block $h^{-1}(b_i, k_i)$; said function h^{-1} being the inverse of said function h; and

- combining sub-blocks t_i derived from said first plurality of output sub-blocks $h^{-1}(b_i, k_i)$ to form said data block B2.

5. A method as claimed in claim 3, wherein said sub-block b_i is derived from said sub-block m_i by bit-wise adding a constant p_i to said sub-block m_i, said constant p_i substantially having equal length as said sub-block m_i.

6. A method as claimed in claim 3, characterised in that said function $h(b_i,k_i)$ is defined by:

$h(b_i,k_i) = (b_i.k_i)^{-1}$, if $b_i \neq 0$, $k_i \neq 0$, and $b_i \neq k_i$

$h(b_i,k_i) = (k_i)^{-2}$, if $b_i = 0$

$h(b_i,k_i) = (q_i)^{-2}$, if $k_i = 0$

$h(b_i,k_i) = 0$, if $b_i = k_i$,

where the multiplication and inverse operations are predetermined Galois Field multiplication and inverse operations.

7. A method as claimed in claim 6, wherein deriving said sub-blocks t_i from said output sub-blocks $h(b_i, k_i)$ comprises bit-wise adding a constant d_i to said output sub-block $h(b_i,k_i)$, said constant d_i substantially having equal length as said sub-block m_i.

8. A method as claimed in claim 7, wherein deriving said sub-blocks t_i from said output sub-blocks $h(b_i, k_i)$ further comprises raising $h(b_i,k_i) \oplus d_i$ to a power 2^i, using said predetermined Galois Field multiplication.

9. A method as claimed in claim 6, wherein deriving said sub-blocks t_i from said output sub-blocks $h(b_i, k_i)$ comprises raising said output sub-block $h(b_i,k_i)$ to a power 2^i, using said predetermined Galois Field (GF) multiplication.

10. A method as claimed in claim 4, wherein said combined data is formed by:

- swapping the sub-blocks t_i and t_{2n-1-i}, for $i = 0$ to $n-1$; and
- concatenating the swapped sub-blocks.

11. A method as claimed in claim 6, wherein said sub-block m_i comprises eight data bits, and wherein said multiplying of two elements b and c of $GF(2^8)$ comprises executing a series of multiplications and additions in $GF(2^4)$.

12. A method as claimed in claim 11, wherein said multiplying of said two elements b and c comprises:

- representing b as $a_0 + a_1.D$ and c as $a_2 + a_3.D$, where a_0, a_1, a_2 and a_3 are elements of $GF(2^4)$, and where D is an element of $GF(2^8)$ defined as a root of an irreducible polynomial $k(x) = x^2 + x + \beta$ over $GF(2^4)$, where β is an element of $GF(2^4)$; and
- calculating $(a_0a_2 + a_1a_3\beta) + (a_1a_2 + a_0a_3 + a_1a_3).D$.

13. A method as claimed in claim 12, wherein β is a root of an irreducible polynomial $h(x) = x^4 + x^3 + x^2 + x + 1$ over $GF(2)$.

14. A method as claimed in claim 6, wherein said sub-block m_i comprises eight data bits, and wherein calculating the inverse of an element b of $GF(2^8)$ comprises performing a series of calculations in $GF(2^4)$.

15. A method as claimed in claim 14, wherein calculating the inverse of said element b comprises:

- representing b as $a_0 + a_1.D$, where a_0 and a_1 are elements of $GF(2^4)$, and where

D is an element of $GF(2^8)$ defined as a root of an irreducible polynomial $k(x) = x^2 + x + \beta$ over $GF(2^4)$, where β is an element of $GF(2^4)$; and

- calculating $(a_0^2 + a_0a_1 + a_1^2\beta)^{-1}((a_0 + a_1) + a_1D)$.

16. An apparatus for cryptographically converting a digital input block into a digital output block; said apparatus comprising:

first input means for obtaining said digital input block;

second input means for obtaining a first key K1;

cryptographic processing means for converting the digital input block into the digital output block; said conversion comprising merging a selected part M1 of said digital input block with said first key K1 and producing a data block B1 which non-linearly depends on said selected part M1 and said first key K1, and where a selected part of said digital output block is derived from said data block B1; and

output means for outputting said digital output block,

characterised in that said cryptographic processing means is arranged to perform said merging by executing a non-linear function g for non-linearly merging said selected part M1 and said first key K1 in one, sequentially inseparable step.

17. An apparatus as claimed in claim 16, wherein said apparatus comprises third input means for obtaining a second key K2, and wherein said conversion comprises:

- splitting said digital input block into said selected part M1 and a second part M2 before performing said merging;
- executing a non-linear function g^{-1} to merge said second block M2 with said second key K2 in one, sequentially inseparable step, producing a data block B2 as output; said non-linear function g^{-1} being the inverse of said non-linear function g; and
- forming combined data from data in said data block B1 and in said data block B2; said digital output block being derived from said combined data.

18. An apparatus as claimed in claim 16, wherein said merging step comprises the steps of:

- splitting said selected part M1 in a first plurality n of sub-blocks $m_0,..,m_{n-1}$ of substantially equal length;
- splitting said first key K1 in said first plurality n of sub-keys $k_0,..,k_{n-1}$, substantially having equal length, the sub-key k_i corresponding to the sub-block m_i, for $i = 0$ to $n-1$;

and

- separately processing each of said sub-blocks m_i by executing for each of said sub-blocks m_i a same non-linear function h for non-linearly merging a sub-block b_i derived from said sub-block m_i with said corresponding sub-key k_i in one, sequentially inseparable step and producing said first plurality of output sub-blocks $h(b_i, k_i)$; and

- combining sub-blocks t_i derived from said first plurality of said output sub-blocks $h(b_i, k_i)$ to form said data block B1.

19. An apparatus as claimed in claim 18, characterised in that said function $h(b_i,k_i)$ is defined by:

$$
\begin{aligned}
h(b_i,k_i) &= (b_i.k_i)^{-1}, && \text{if } b_i \neq 0,\ k_i \neq 0, \text{ and } b_i \neq k_i \\
h(b_i,k_i) &= (k_i)^{-2}, && \text{if } b_i = 0 \\
h(b_i,k_i) &= (b_i)^{-2}, && \text{if } k_i = 0 \\
h(b_i,k_i) &= 0, && \text{if } b_i = k_i,
\end{aligned}
$$

where the multiplication and inverse operations are predetermined Galois Field multiplication and inverse operations.

20. An apparatus as claimed in claim 19, wherein said sub-block m_i comprises eight data bits, and wherein said multiplying of two elements b and c of $GF(2^8)$ comprises:

- representing b as $a_0 + a_1.D$ and c as $a_2 + a_3.D$, where a_0, a_1, a_2 and a_3 are elements of $GF(2^4)$, and where D is an element of $GF(2^8)$ defined as a root of an irreducible polynomial $k(x) = x^2 + x + \beta$ over $GF(2^4)$, where β is an element of $GF(2^4)$; and

- calculating $(a_0a_2 + a_1a_3\beta) + (a_1a_2 + a_0a_3 + a_1a_3).D$;

and wherein calculating the inverse of an element b of $GF(2^8)$ comprises calculating $(a_0^2 + a_0a_1 + a_1^2\beta)^{-1}((a_0 + a_1) + a_1D)$.